Rough-Fuzzy based Scene Categorization for Text Detection and Recognition in Video

Sangheeta Roy[a], Palaiahnakote Shivakumara[a], Namita Jain[b], Vijeta Khare[a], Anjan Dutta[c], Umapada Pal[c], Tong Lu[d]

[a]*Department of Computer System and Information Technology, University of Malaya, Kuala Lumpur, Malaysia. Email: 2sangheetaroy@gmail.com, shiva@um.edu.my, kharevijeta@gmail.com.*
[b]*Machine Intelligence Unit, Indian Statistical Institute, Kolkata, India. Email: namita.saket@gmail.com*
[c]*Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India. Email: umapada@isical.ac.in, duttanjan@gmail.com*
[d]*National National Key Lab for Novel Software Technology, Nanjing University, Nanjing, China. Email: lutong@nju.edu.cn*

Abstract

Scene image or video understanding is a challenging task especially when number of video types increases drastically with high variations in background and foreground. This paper proposes a new method for categorizing scene videos into different classes, namely, Animation, Outlet, Sports, e-Learning, Medical, Weather, Defense, Economics, Animal Planet and Technology, for the performance improvement of text detection and recognition, which is an effective approach for scene image or video understanding. For this purpose, at first, we present a new combination of rough and fuzzy concept to study irregular shapes of edge components in input scene videos, which helps to classify edge components into several groups. Next, the proposed method explores gradient direction information of each pixel in each edge component group to extract stroke based features by dividing each group into several intra and inter planes. We further extract correlation and covariance features to encode semantic features located inside planes or between planes. Features of intra and inter planes of groups are then concatenated to get a feature matrix. Finally, the feature matrix is verified with temporal frames and fed to a neural network for categorization. Experimental results show that the proposed method outperforms the existing state-of-the-art methods, at the same time, the performances of text detection and recognition methods are also improved significantly due to categorization.

Keywords: Rough set, Fuzzy set, Video categorization, Scene image classification, Video text detection, Video text recognition.

1. Introduction

Due to the recent trend of urbanization such as smart city and digital city developments, new devices are developed for capturing a variety of videos without many constraints [1, 2]. As a result, the explosive proliferation of multimedia content available on broadcast and internet has led to the increasing need for its ubiquitous access at any time or any-where. For instance, photo sharing websites (e.g., Flickr) host billions of images with thousands of uploads every minute, similarly video sharing websites (e.g., YouTube) host millions of videos with hours of new videos uploaded every minute [3]. Therefore, when we have such lengthy voluminous video programs, it needs to access interesting parts and skip less interesting parts of videos to save viewers time and cost of data downloading especially when viewers are in travel [3]. It would be attractive if viewers can access and view the content based on their own choices. This makes the problem more challenging and interesting because user interests are often unpredictable. Moreover, one can expect diversified data collections of large sizes. For example, a large size dataset may contain videos of different classes like (i) Sports - which contain court scenes of different sports with type text, (ii) Defense - which contains army vehicle scenes with multi-oriented texts in complex background, (iii) Weather - which contains reports of different regions with texts of different fonts, (iv) e-Learning - which contains slides, lecture notes

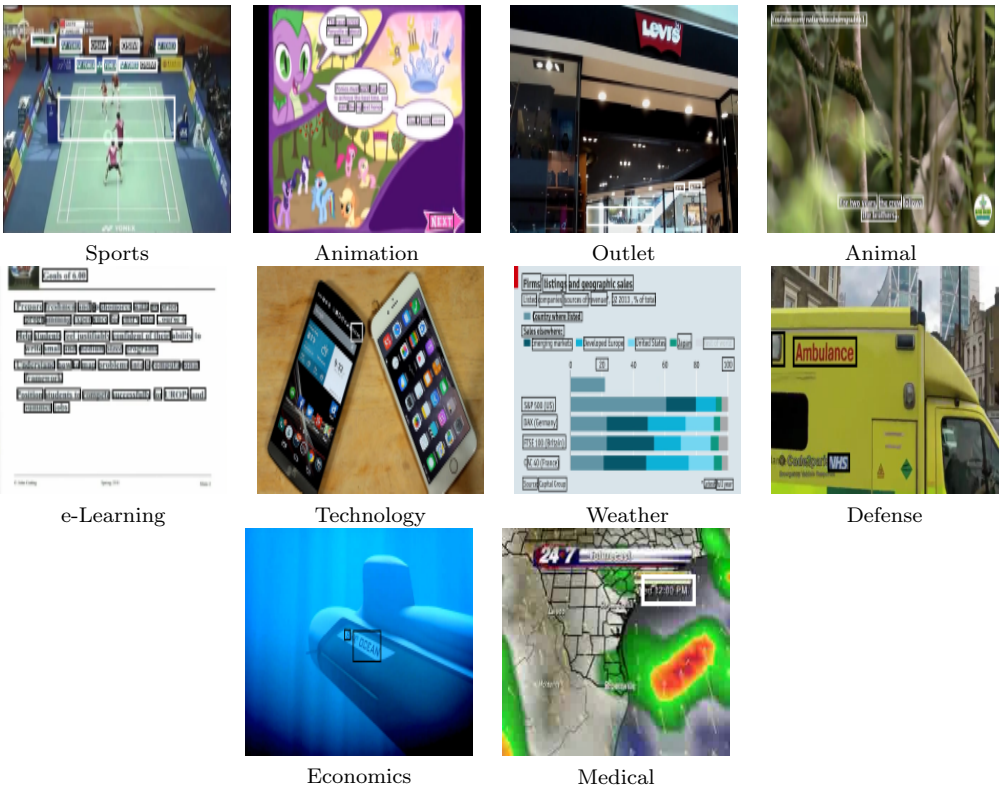


Figure 1: Text detection results (marked by rectangle) for frames of different scene type videos by the existing method in [6] (for better visibility of the images see the PDF file.)

or scanned books with large fonts and text appearance variations, (v) Medical - which may contain vehicles in different background scenes with multi-oriented scene text, (vi) Technology - which contains devices with different scene backgrounds, (vii) Outlet - which contains shops in different malls or supermarkets with different types of scene texts, (viii) Animal Planet - which contains animals with caption texts, (ix) Economics - which contains different charts with different font types, (x) Animation - which contains movies or stories for kids with lots of fancy and animated texts, etc. Sample video frames of each mentioned class can be seen in Fig. 1, where one can guess that each video has its own natural scenes and characteristics.

Therefore, labeling particular videos from such diversified and huge databases is challenging [4]. To solve this problem, many methods have been developed in the field of content based image retrieval (CBIR). Although these methods work well, they still have inherent limitations to retrieve videos due to the gap between low level and high level features in generating semantics [2, 5]. To overcome semantic issues, if a video contains text information, text detection and recognition methods can be used to label it using text information. However, it is noted from literature that these methods achieve good results for a particular scene type video but show lower performances for heterogeneous scene type videos. One such example is shown in Fig. 1, where we can see that text detection results obtained by the method in [6] is robust to text detection in natural scene images but gives inconsistent results on heterogeneous data due to large variations in video frames. Fig. 1 also shows that each scene type image has different background complexity, which affects adversely to study object patterns in images, including text patterns. The same conclusions are true for recognition results [7, 8].

There are two ways to overcome the above issues: one way is to develop a universal method which is complex and not advisable for text detection and recognition, and the other way is to identify classes (different scene type videos) according to the complexity of videos. The latter way can be useful for selecting an appropriate OCR/classifier [8], and then we can modify the existing methods to achieve good

results. Therefore, inspired by the work proposed in [9] where it is shown that identifying text of different complexities (scripts) in images helps in achieving better results for recognition, in this work we propose a new categorization method for identifying different scene type videos to enhance the performance of text detection and recognition.

2. Related Work

Since the focus of the work is to categorize different scene type images containing text information to enhance text detection and recognition performance, we review the methods on scene classification, text detection and recognition in video images here.

When we look at the literature on news video classification [10], we notice that most of the methods use more than one media, namely, audio, caption text or visual content, for classification. Jasper et al. [4] proposed a real time visual concept for classification. Ewerth et al. [11] proposed long term incremental web supervised learning of visual concepts via random savannas. Chen et al. [12] proposed automatic training image acquisition and effective feature selection from community contributed photos for facial attribute detection. These methods explore descriptors and classifiers for the classification of images or videos. The methods require objects in videos or images to achieve better results. Unlike these methods, the proposed method does not expect objects with specific shapes and explores temporal information for classifying different scene type videos containing text information.

Recently, some methods explored deep learning and convolutional networks for video classification because they believe that deep learning framework is capable of solving the complex video classification problem [13–22]. It is noted from the review of the above methods that most methods focus on particular data types and deep learning frameworks are designed according to requirements [23, 24]. It is not clear that whether the classification is useful and whether the methods can work for different scene type videos where one cannot expect objects with specific shapes. Besides, none of the methods focuses on scene type videos containing texts for categorization and further uses text information for validating classification through text detection and recognition.

At the same time, when we look at the literature on text detection and recognition in scene type videos as mentioned in the Introduction Section 1, most of the methods focus on a particular type of video for achieving better results. For instance, it is noted from [25–37] that despite these methods address complex issues such as images with low contrast and complex background with multi-oriented texts, they report inconsistent results for different scene type videos or images. The same conclusion can be drawn from the methods developed recently by exploring deep learning and convolutional neural networks, which work well for the images affected by different causes [38–41] However, setting or predicting a deep learning framework and its parameters is not so easy when we have different scene type videos with large variations in background and foreground complexities [23, 24]. In addition, most of the methods expect multiple objects with specific shapes for feature extraction. This is not necessarily true for the scene type videos considered in this work. Therefore, in light of above discussions on classification and text detection/recognition, we can conclude that the considered scene type videos pose open challenges for categorization due to variation in background complexities and foreground complexities. In addition, it is also noted from text detection and recognition that none of the methods reports satisfactory results for the considered scene type videos. These factors motivated us to propose a new method based on rough-fuzzy combination for scene type video categorization to enhance the performance of text detection and recognition.

The key contributions of the proposed method are as follows. (1) we explore the combination of rough and fuzzy to classify irregular edge components into particular groups in a new way, (2) we propose a membership function in a different way to define shapes of edge components that have irregular edge patterns, (3) we explore gradient directions of pixels across planes, which are given by the directions, to find the relationship between strokes of edge components, and (4) further, temporal information is used for adding stability to features.

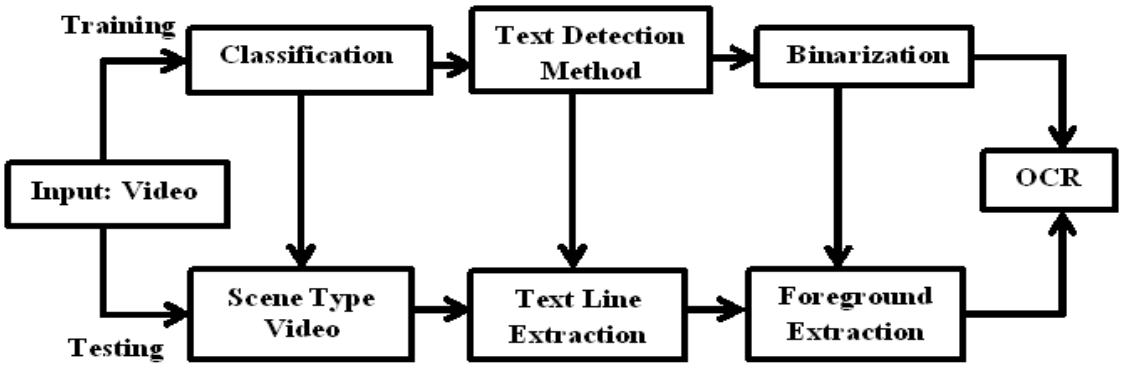


Figure 2: Unified frame work of the proposed method.

3. Proposed Method

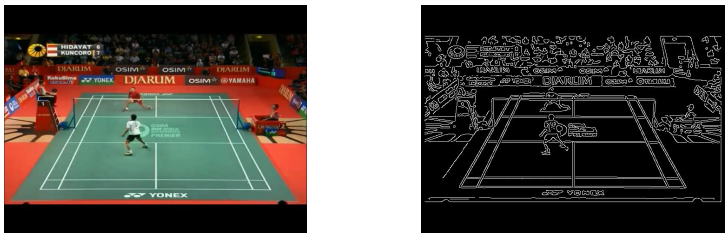
In this work, we propose a unified system as shown in Fig. 2, where the classification step identifies video types, the text detection step extracts text lines from video frames, the binarization step extracts foreground from text lines, and finally OCR recognizes texts. Fig. 2 shows that parameters for respective steps are derived automatically with the help of training samples. Since the main goal of this work is video categorization, we focus on the classification method and use text detection and recognition performances for validating the proposed classification.

For a given scene type video, in this work, we extract key frames and neighboring temporal frames for classifying scene type videos. For each frame, the proposed method obtains its Canny edge image, which gives edge components with their structures [34]. The advantage is that it saves a large number of computations as it helps in extracting features at component level rather at pixel level. In order to classify each input video frame as a particular class according to the nature of its content, we propose a new combination of rough set and fuzzy logic to group edge components as Line, Rectangle, Square, Parallelogram, Circle, Loop, Ellipse and Trapezium based on geometric shapes of edge components to extract local information of the frame. Since the considered video categorization problem is complex, one can expect uncertainty in defining shapes of edge components. Therefore, to deal such situations, we introduce rough set to estimate lower and upper boundary approximations [42], which give boundaries for extracting shapes of edge components. Due to foreground and background variations, an approximated shape by rough set may overlap with other shapes of edge components. This leads to confusion or uncertainty. Therefore, motivated by [43] where it is shown that fuzzy logic for recognizing elementary geometric shapes is useful in improving object recognition, we introduce fuzzy logic to recognize geometric shapes [43, 44]. This step outputs eight groups according to the shapes defined by rough set and fuzzy combination for the given video frame. The eight groups are empirically determined by studying shapes of edge components for different classes.

It is true that gradient directions of edge pixels represent stroke direction distribution, which in turn provide a vital clue for extracting shapes of edge components [45]. Therefore, we divide each group into several planes according to the gradient direction of pixels of edge components in each group. For each plane, we further propose to extract correlation and covariance features using gradient values to encode statistical and spatial correlation between stroke directions. Features are extracted for all the eight groups by this way. Furthermore, to add stability to these features, we explore temporal frames. Finally, the feature matrix is passed to a neural network classifier for frame classification.

3.1. Edge Components Detection

For the sample video frame chosen from sports class as shown in Fig 3a, the proposed method obtains Canny edge image as shown in Fig. 3b, where we can see edge components preserve the structure of components. It is also observed from Fig. 3b that the edge components which represent background and foreground (text) have different shapes, such as rectangles, loops, ellipses, parallelograms, circles, trapeziums, squares



(a) Sports frame (b) Canny edge image

Figure 3: Edge component detection for the sample sports input frame.

and lines. Among different shapes, rectangles and lines are more prominent due to the presence of courts. This observation leads to propose a new method for classifying those components into several groups to find the relationship among them for video classification.

3.2. Rough-Fuzzy for Components Grouping

As discussed in the earlier part of this section, we explore the combination of rough set and fuzzy logic for grouping edge components of each frame of each class into different geometric shapes, namely, Rectangle, Parallelogram, Trapezium, Circle, Ellipse, Loop and Line. The identification of Line is done using the projection of pixels on the principal component axis, while the identification as Loop is done by checking whether a component is split into several sub-components after removing a few pixels. Since Square is a special case of rectangle, we use the same rectangle steps for the identification of Square. The identifications of Rectangle, Parallelogram, Trapezium, Circle and Ellipse are done as follows: For each edge component C as shape S, we construct the smallest object of shape S which covers C (we name it the mask of component C and denote it by M_{CS}). We expect this mask precisely overlaps C for an ideal shape of edge component. However, this is not always true for real edge components due to irregular shapes and disconnections, where uncertainties are expected. Therefore, we propose to match the component with the boundary of another set R_{CS}, such that R_{CS} is an approximation of M_{CS}. Approximation using rough set allows us to ignore small errors and decide whether the component under consideration belongs to a given class S. However, even if this component does not belong to S, we may want to decide how similar it is to class S. This is done by using the proposed fuzzy membership function dicussed below. Fig. 4a shows an ideal component. In Fig. 4b the boundary of rough set R_{CS} is shown with white color and the interior of R_{CS} is shown with blue color. Note that the component is marked in green color. Fig. 4c shows the precise overlap of R_{CS} boundary and component boundary. The advantage of the combination of rough approximation and fuzzy membership to define irregular shapes of edge components can be seen in Fig. 5, where for the component in (a) we can see its boundary and interior of R_{CS} in (b). Note that the overlap between R_{CS} and the component as shown in (c) is partial for this component. From now, we refer to this approximation R_{CS} as the mask. The boundary of this mask is defined using rough set theory as follows.

Formally, we can define rough set with lower and upper boundaries approximation as follows. Let X be the reference set ($X \subset U$ is the reference set, i.e., the set we want to approximate, while U is the universe and refers to all the pixels in the image). Lower approximation of X is the region where all the data definitely belong to X. Upper approximation is the region such that no point outside this region belongs to X. The difference between upper and lower approximations is defined as the boundary of the rough set. This is the region where some points belong to X and some do not. It is illustrated in Fig. 5, where (a) gives the sample edge component, (b) shows the lower approximation (blue color) and the upper approximation (white color + blue color), and (c) is the result of actual overlap between the estimated rough set boundary and the edge component boundary. Formal definitions of both approximation and boundary region are given by [46] as follows:

(a) Edge component (b) Granulation (c) Approximation

Figure 4: Illustrating rough approximation for an ideal edge component, where the component and its mask boundary overlap completely. (a) represents the edge component, (b) represents its mask boundary estimated as white region, while the interior of the mask is shown in blue color, and (c) is the overlapping region between the component and its mask boundary in white color.

$$\begin{aligned} &\textit{R\_lower approximation of X } R_{*}\left(x\right) = \bigcup_{x \in U} \left\{R\left(x\right) : R\left(x\right) \ \subseteq X\right\} \\ &\textit{R\_upper approximation of X } R^{*}\left(x\right) = \bigcup_{x \in U} \left\{R\left(x\right) : R\left(x\right) \ \cap X \neq \emptyset\right\} \\ &\textit{R\_boundary of X } RN_{R}\left(x\right) = R^{*}\left(x\right) - \ R_{*}\left(x\right) \end{aligned} \tag{1}$$

We define boundary R_{CS} as the set of all the points p such that the neighborhood of p contains some points belonging to mask M_{CS} and some points belonging to $M_{CS}{}'$ as stated in equation 2.

$$\textit{boundary}\left(R_{CS}\right) = \left\{p : N\left(p\right) \cap \ M_{CS} \neq \emptyset \ \text{ and } \ N\left(p\right) \cap M_{CS}{}' \neq \emptyset\right\} \tag{2}$$

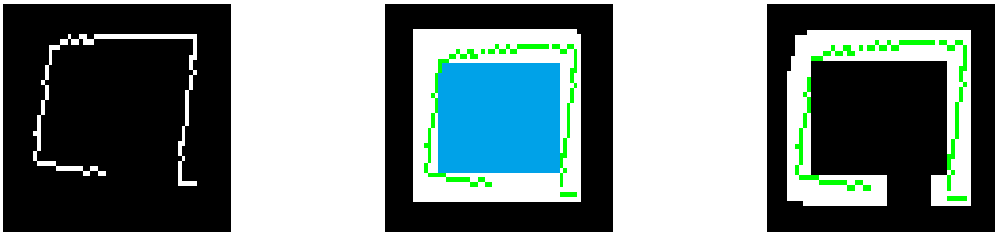
where the lower approximation of R_{CS} is $R_{CS*} = M_{CS} - boundary(R_{CS})$ and the upper approximation is $R^{*}_{CS} = M_{CS}{}' - boundary(R_{CS})$; Here $M_{CS}{}'$ denotes the complement of the set M_{CS}, while *N(p)* represents the neighborhood of point p. The neighborhood is given by the open disc of chosen radius r. Here, the lower approximation is the set of all the pixels which definitely belong to the estimated shape. This is the interior of the estimated component as shown in Fig. 5b. The upper approximation is the set of all the pixels which may belong to the estimated shape. This region is the complement of exterior of the component. Boundary is the region which is close to both interior and exterior regions of the estimated shape. Rough sets allow us to identify a component to be of a perfect shape if interior and exterior regions match perfectly. This is done by using a thicker boundary region rather than actual outline of the component for comparison as shown in Fig. 5c. To estimate boundary approximation for edge component C, the proposed method checks weather all the pixels lie in boundary(R_{CS}) and those pixels which are the neighbors of the component. If a component satisfies both the conditions, it is considered as the component roughly like shape S.

If *boundary* (R_{CS}) overlaps with component C completely, it is said to be the component that is exactly like S as shown in Fig. 4. Otherwise, we need to estimate the degree of overlap information to find the closeness between the boundary and the component as shown in Fig. 5. Let the ratio of component pixels close to mask boundary and the total number of component pixels be u_c, and the ratio of mask boundary pixels close to component pixels and the total number of mask boundary pixels be u_m. In order to find the final value which indicates how close C is to shape S, we apply a Z shaped fuzzy membership function to $1 - min(u_c, u_m)$ as defined in [47]. Z shape fuzzy membership function is a spline based one as defined in

equation 3 and illustrated in Fig. 6a.

$$z(x,s,t) = \begin{cases} 1 & if\ x \leq s \\ 1 - 2(\frac{x-s}{t-s})^2 & if\ s < x \leq \frac{s+t}{2} \\ 2(\frac{x-t}{t-s})^2 & if\ \frac{s+t}{2} < x \leq t \\ 0 & if\ t < x \end{cases} \tag{3}$$

If the membership function gives 1, C is an ideal example of the shape S [47]. Here we irrespective set the value as 1 for all the values which are less than s, and 0 for all the values which are greater than t. However, the values always lie between s and t for real edge components. In this work, we determine the values for s and t experimentally according to the defined geometrical shapes.



(a) Edge component (b) Granulation (c) Approximation

Figure 5: Rough set is defined for the edge component of the sports frame where edge component boundary and mask boundary does not match completely. (a) is edge component with loss of information, (b) shows mask boundary estimated for the component as white region and interior of the mask is shown in blue color and (c) Overlapping region between component and mask boundary in white color.

Apart from s and t required for applying rough set and fuzzy logic, other parameters a, b, r, θ are required for recognizing shapes of different edge components according to groups as follows:
Rectangle: Let the height and width of the rectangle bounding box be $2a$ and $2b$, respectively, and the centroid be (x_0, y_0). Thus *Rectangle* can be defined as in equation 4:

$$y \geq y_0 - a,\ y \leq y_0 + a,\ x \geq x_0 - b,\ x \leq x_0 + b \tag{4}$$

Square: If the given component is detected to be rectangle, check whether both the sides of the bounding box are nearly equal. *Parallelogram*: A right tilted parallelogram will be defined as in equation 5:

$$\begin{gathered} y \leq \tan\theta\ (x - x_0 + b) + y_0 - a, \\ y \geq \tan\theta\ (x - x_0 - b) + y_0 + a, \\ y \geq y - a, y \leq y_0 + a \end{gathered} \tag{5}$$

Similarly, a left tilted parallelogram is defined as in equation 6:

$$\begin{gathered} y \geq -\tan\theta\ (x - x_0 + b) + y_0 + a, \\ y \leq -\tan\theta\ (x - x_0 - b) + y_0 - a, x \geq x_0 - a, \\ y \geq y - a, y \leq y_0 + a \end{gathered} \tag{6}$$

Regular trapezium: A regular trapezium can be defined as in equation 7

$$y \leq \tan\theta\ (x - x_0 + b) + y_0 - a,\ y \leq -\tan\theta\ (x - x_0 - b) + y_0 - a,$$

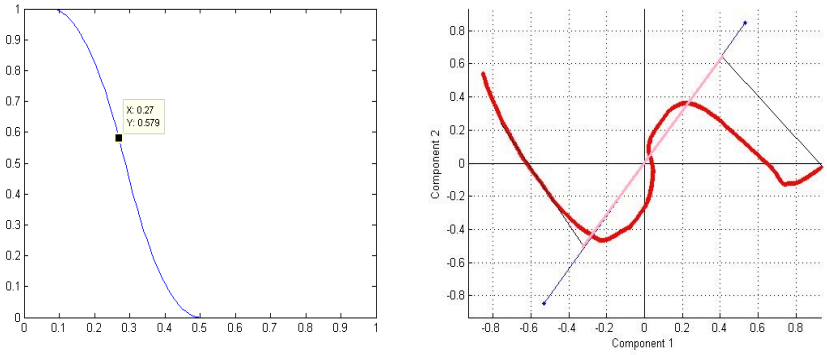
$$y \geq y - a, y \leq y_0 + a \tag{7}$$

Circle: The proposed method finds the centroid and the pixel which is the farthest from the centroid. The distance between the centroid and this pixel gives radius r, and centroid (x_0, y_0) gives the center of the edge component. Therefore, the mask given by $(x - x_0)^2 + (y - y_0)^2 \leq r^2$ can be calculated using these parameters. The points on this mask can be generated as $(x_0 + r\cos\theta,\ y_0 + \ r\ sin\theta\ )$ for varying values of θ from 0 to 2π.

Ellipse: Given length $2a$ and width $2b$ and the location of centroid $(x_0,\ y_0)$, an ellipse defined by $\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} \leq 1$ gives the required ellipse. The points on this ellipse can be generated as $(x_0 + a\cos\theta,\ y_0 + \ b\ sin\theta\ )$ for varying values of θ from 0 to 2π.

Loop: Suppose the given component contains n pixels, we choose $n/10$ equidistant pixels from the component. For each pixel, we delete those pixels which are at a distance of two pixels or less from the chosen pixel. After removing these pixels, if the remaining edge component is still a connected component, then the chosen pixel is a part of a close loop. Otherwise it is a part of an open component. Repeat this procedure $n/10$ times, each time on the original image of the component. As long as the resultant edge component remains a connected component, the proposed method estimates the percentage of pixels which are a part of a loop. Note that the mask algorithm and Z shaped analysis are not used for loop test.

Line: We plot the principal component axis for an edge component as shown in Fig. 6b, where it can be seen that the principal axis is given by principal component analysis and its projections. The value of area obtained is scaled by the square of the length of the principal axis segment corresponding to it. The value of the area obtained by the above procedure can lie in the interval $[0, \infty)$. The values obtained are fed to Z shape fuzzy membership function.



(a) (b)

Figure 6: Fuzzy membership functions for classification of edge components according to shapes. (a) Z Fuzzy membership function for classification of edge components according to shapes. X axis denotes the original value calculated and Y axis shows resulting membership value and (b) Membership function for line and curve shaped edge components.

The values obtained from the mask overlap algorithm and the line function are fed to Z shape fuzzy membership function. Fig. 6a shows this function with parameter values $s = 0.1$ and $t = 0.5$. The values of parameters s and t used for fuzzy filter are $s = 0.075$ and $t = 0.5$ for each of rectangle, parallelogram, trapezium, circle and ellipse. For line, $s = 1e - 3$ and $= 0.02$ are considered.

The above process of recognizing shapes by fixing mask boundary and estimating parameters of edge components work well for those edge components having zero degree orientations with horizontal. However, in case of irregular shaped edge components, one cannot expect all the time edge components without any tilt or orientation. To overcome this problem, we propose to use Principal Component Analysis (PCA) for estimating angles of edge components with respect to X axis. Edge components are then rotated by this angle, which results in edge components with zero orientation. In other words, we propose to use PCA to check the orientation of each input edge component before applying the boundary approximation

for them. It is true that when an edge component is rotated by an angle, we can expect tiny distortion. However, the proposed rough set and fuzzy combination takes care of such tiny distortion affected by rotation conversion. Orientation checking using PCA is good for those shapes like Rectangle, Parallelogram, Ellipse and Trapezium. Since Square is treated as a special case of rectangle, orientation checking is similar to rectangle. On the other hand, for Circle, orientation checking is not necessary as this shape does not affect the above process of recognizing shapes. For Loop and Line, the proposed method uses an iterative procedure and the projections on the principal axis as presented earlier, respectively. It is noted that these two procedures are invariant to rotation. Sample illustration of the process of recognizing shapes, which involves rotating edge components to zero orientation, estimating mask boundary and finding overlapping region between component boundary and mask boundary for rotated/tilted edge components are shown in Fig. 7.

(a) Oriented rectangle (b) Converted to zero orientation (c) Mask boundary (d) Approximation

(e) Oriented parallelogram (f) Converted to zero orientation (g) Mask boundary (h) Approximation

Figure 7: Fixing mask boundary and recognizing shapes for rotated or tilt edge components. Note: for the purpose of visualization, we perform morphological operation for the edge components. In case of mask boundary results, blue color denotes interior of mask boundary. In case of approximation, green color denote edge component boundary, while the white region represents the overlapping between edge boundary and mask boundary.

Sample edge components for grouping according to the shapes defined above are shown in Fig. 8, where we can see that the edge components are classified according to the definitions of geometrical shapes. As mentioned in Section 3.1, we prefer to use Canny edge detector because it has the ability to preserve structures of edge components and to generate fine edges for both low contrast and high contrast frames, which we considered in this work as shown in Fig. 8. From Fig. 8, we can see edge components for all the groups in spite of low contrast input video frames. To know the effect of Canny edge detector, we compare it with the Sobel edge detector for the same input frame to classify edge components into respective groups as shown in Fig. 9, where we can notice that a few pixels are missing in the groups compared to the groups of Canny edge detector. This shows that Sobel edge detector loses some times edges for low contrast video frames, which leads to poor performances. Experimental results are provided in Section 4 to support the statement.

3.3. Intra Plane Feature Extraction

For each group given by the above presented method in the previous section for video frames of every class, we explore gradient direction and values for extracting distinct features to classify frames. Motivated by the work proposed in [45] for recognizing handwriting characters, where it is shown that stroke distribution provides the vital clue for recognizing different handwriting styles of characters, we explore the gradient directions by distributing pixels into a number of planes according to gradient direction to find local distribution of pixels. It is illustrated in Fig. 10, where (a) shows the sample image of Line group, its gradient image, and gradient directions, and (b) shows pixel distribution into a number of planes according to gradient angle, which generally varies from -180 to + 180. This process results in angular planes for each group. For each angular plane, we apply k-means clustering on the gradient values of pixels to obtain

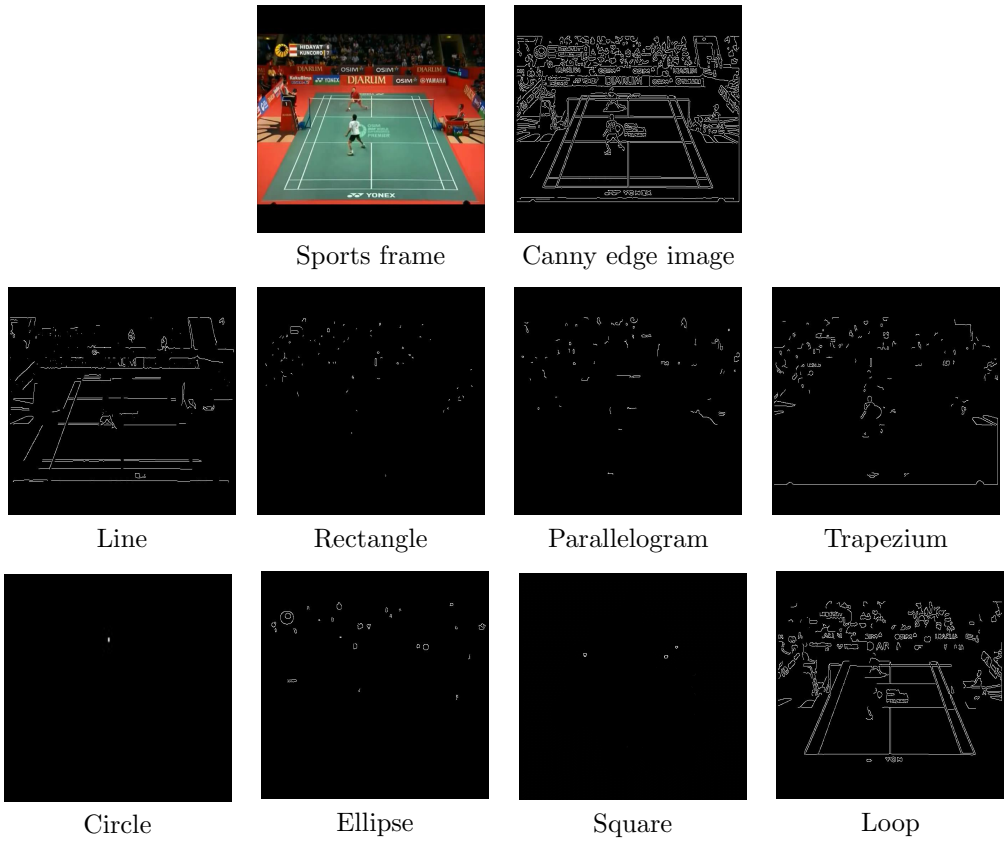


Figure 8: Sample components grouping based on shape analysis for the Canny edge image of the sports frame using rough-fuzzy.

different clusters as shown in Fig. 11, where we can see clusters with different colors for P1 plane in Fig. 10b. This helps in finding the relationship between the pixels in each plane. Here the relationship is defined as how the pixels are close to each other in terms of contrast. Then the proposed method computes the mean, median and standard deviations for each cluster, which gives 3 features for each plane. The relationship among the pixels in each plane, which we call intra plane, is encoded by covariance and correlation as defined in equation 8 and equation 9. Inspired by the work in [45], we propose the same features for feature vectors of the planes. This process of feature extraction results in a 3600 dimensional feature matrix for each input frame.

$$cov\left(X,Y\right)=\frac{\sum_{i=1}^{n}\left(X_i-\overline{X}\right)\left(Y_i-\overline{Y}\right)}{n} \tag{8}$$

where X and Y are real valued random variables.

$$R\left(X,Y\right)=\frac{cov\left(X,Y\right)}{\sqrt{cov\left(X,X\right)*cov\left(Y,Y\right)}} \tag{9}$$

The feature extraction process is formulated as follows. Let the feature matrix (M) be of size of $p\times 15$, where p is the number of the total angular planes in the frame, while 15 is the dimension of feature vectors ($k=5$ clusters and their 3 features). To apply covariance according to equation 8, we calculate the mean and deviation for each feature vector. This yields a matrix of size $p\times 15$ using equation 9

$$mv_{p\times 15}=M-II^{'}M\left(1/p\right) \tag{10}$$

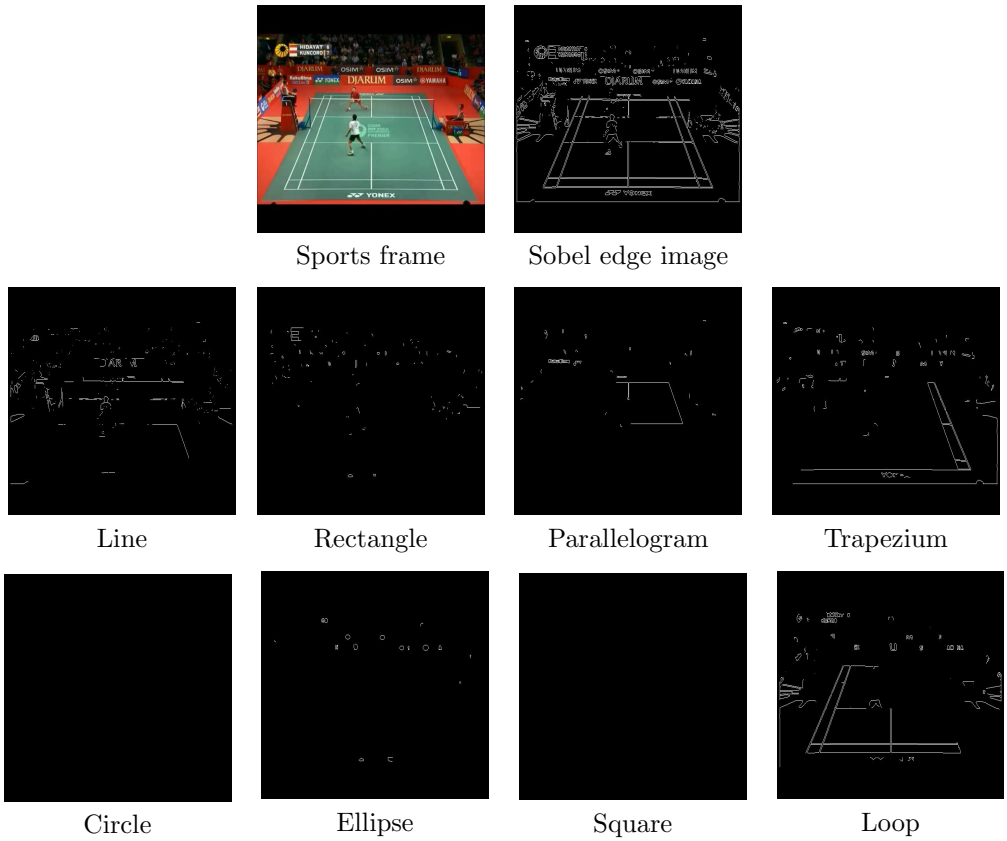


Figure 9: Sample components grouping based on shape analysis for Sobel edge of the input sports frame with rough-fuzzy.

where I is a $p \times 1$ column vector of ones, and I' is the transposed matrix of it. We then transpose $mv_{p\times 15}$ and multiply it with the original matrix M using the following equation 11

$$MV_{15\times 15} = mv'_{p\times 15} \times mv_{p\times 15} \tag{11}$$

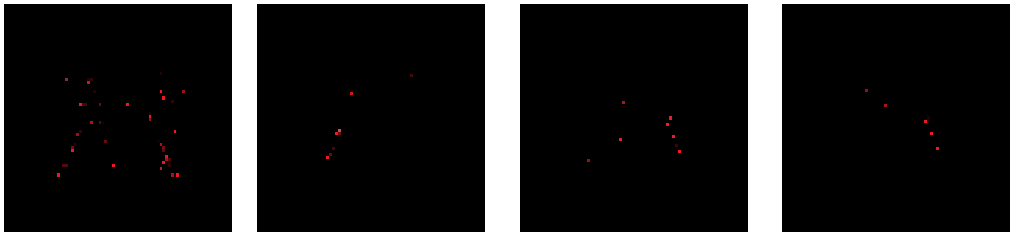
It generates a feature matrix of size 15×15. And finally, covariance matrix is obtained by dividing with the number of planes p as defined in equation 12

$$cov\_M_{15\times 15} = MV_{15\times 15}/p \tag{12}$$

In the same way, the correlation feature matrix of size 15×15 is reckoned using matrix multiplication and division according to equation 9. As a result, for the angular plane P1 in Fig. 10b, the proposed method gives 5 clusters as shown in Fig. 11. Therefore, the proposed method obtains 225 features using covariance and 225 features using correlation for each group. For 8 groups given by Section 3.2 of the frame in Fig. 3a, the proposed method obtains 3,600 features, which we call intra plane features as it uses each plane separately. Here the value of k is 5 for intra plane features.

In order to find the value for k automatically, we propose the following procedure. We choose 100 samples from each of the 10 classes randomly, which gives 1,000 frames for determining the value of k automatically. For each frame, the proposed method obtains a number of planes as discussed above for the Canny edge image of the frame. For each plane, we apply k-means clustering with k=n, which gives different numbers of clusters (the value of k) for each plane as shown in Fig. 12a for frame 1 (F1) and frame 2 (F2), where we can see different k values for different planes. To find the common cluster number which represents the planes of the frame, we perform histogram operation for k values of planes as shown in Fig. 12b, which we

(a) Line group, Gradient and Direction image.



(b) Planes division: P1-Angle: 180, P2-Angle: -162, P7-Angle:- 135, P16-Angle:-19, according to gradient directions (from left to right).

Figure 10: Example of plane generation according to gradient direction to extract structural features (Best viewed in PDF).

call a local histogram. The propsoed method chooses the value which gives the highest peak as k value at plane level that should be represnted at frame level. In order to find k value at frame level, which reprsents the whole database, we perform the same histogram operation on k values chosen for each sample frame to choose the value that contributes to the highest peak as shown in Fig. 12c as k value for intra plane features, where we can see k=5 gives the highest peak.

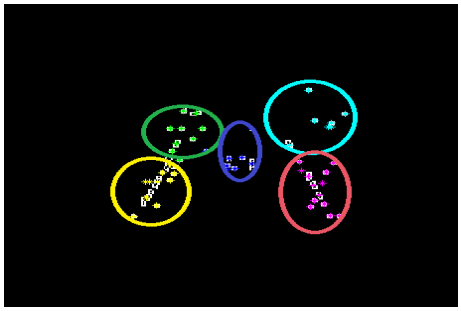
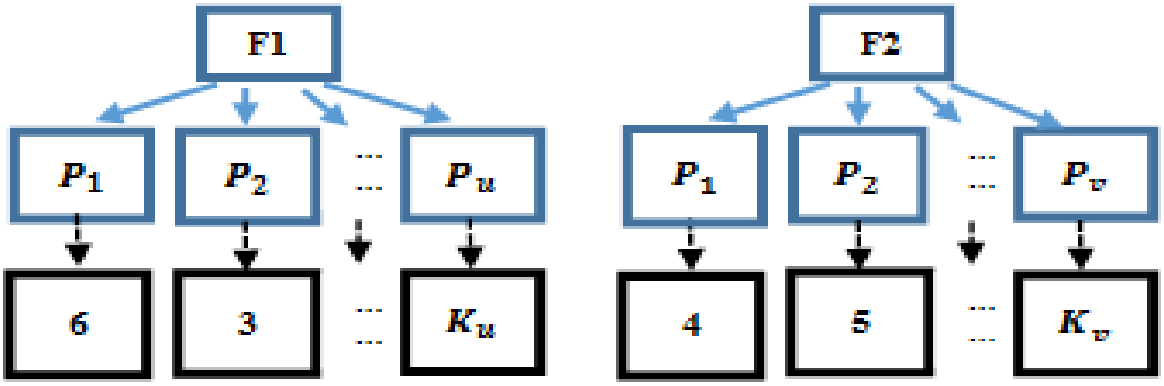


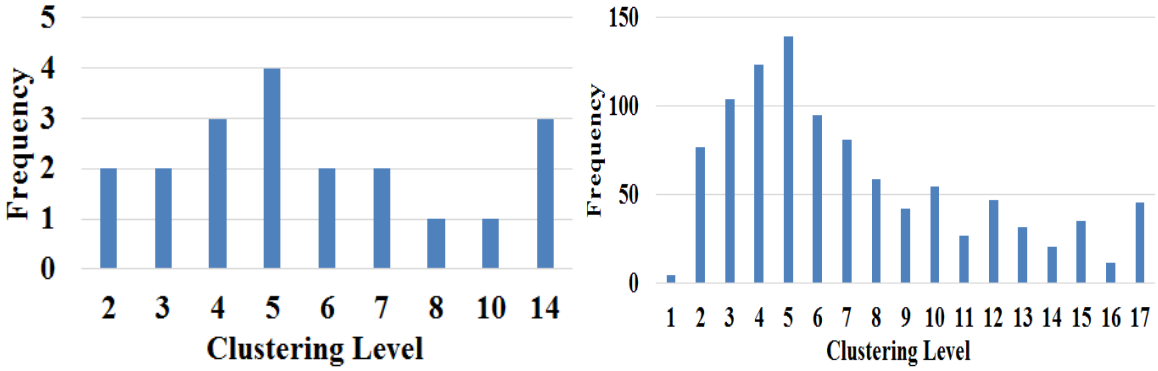
Figure 11: Different clusters for plane P1 in Fig. 10b.

3.4. Inter Plane Feature Extraction

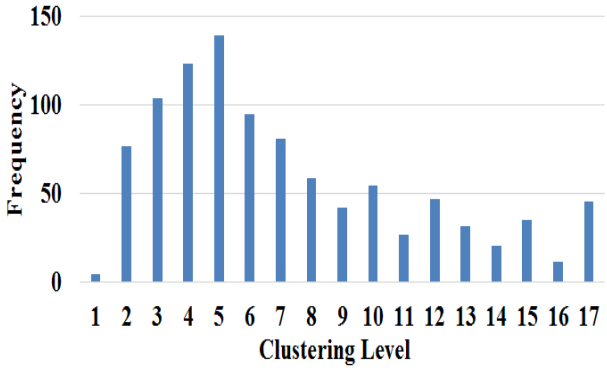
We extract intra plane features in the previous section to find the relationship between plane pixels. This leads to extract features across planes to strengthen feature extraction to solve the complex scene type video categorization problem. The number of clusters for planes is determined with the predefined samples. For each plane, we apply k-means clustering to obtain clusters as shown in Fig. 13, where we can see 8 clusters for plane P1. For the second plane also, the proposed method obtains clusters. To find the relationship between inter planes, we extract gradient values across the planes corresponding to the edge components in the clusters. For those gradient values, we plot a histograms to find the value which contributes for the highest peak as the feature of the particular cluster. This gives a feature vector for one cluster. In the



(a) Clusters values with k-n for the planes of sample frames, F1, F2 and so on.



(b) Local histogram for choosing k values at plane level in unsupervised clustering.



(c) Global histogram for choosing k values at frame level.

Figure 12: Determination of the value for k automatically for intra plane features.

same way, the proposed method obtains feature vectors for other clusters, which forms a feature matrix. The covariance and correlation are estimated for the feature matrix as mentioned in the previous section, which gives 64 features (8×8) for each group given by the method presented in Section 3.3. In total, since k is 8, the number of features would be $64\times\ 8 = 1024$ for the each frame. As discussed in the previous section, to determine the value for k automatically, we also propose the following procedure using the same 1, 000 samples. The proposed method obtains Canny edge image (C) for the samples as shown in Fig. 14a, then it applies k-means clustering with k=m for each edge image as shown in Fig. 14a, where we can see different k values for different frames. To choose the value for k, we perform histogram on k values as shown in Fig. 14b, where it is noticed 8 is contributing to the highest peak and hence it is considered as the actual value of k. For the input frame, we extract 3600+1024 = 4624 features for classification. In summary, algorithmic representation for inter plane feature extraction and determining the value of k are presented below. The steps in training phase describes how to determine the number of clusters k, which is the parameter of k-means clustering using predefined samples. For each sample, the proposed method obtains Canny edge images. Then k-means clustering is applied on all the Canny edge images, which outputs a number of clusters (the value of k) for each sample. To choose k value which represents the whole database, we perform histogram on k values obtained for each sample. The value which contributes to the highest peak is considered as the actual k value of k-means clustering at frame level. Similarly, the steps in testing phase describe how to extract features for testing samples. For each testing sample, the proposed method obtains 8 groups using rough-fuzzy combination method presented in Section 3.2. Each group is divided into angular planes based on gradient direction of components in the group. For each angular plane, the proposed method employs k-means clustering with k (determined earlier) value on each plane, which results in k clusters. The proposed method extracts gradient values across the planes corresponding to edge components in the clusters. For those gradient values, we plot a histogram to find the value which contributes the highest peak as the feature vector of the particular cluster. The covariance and correlation features extracted for the feature vectors and this results in a feature matrix for classification.

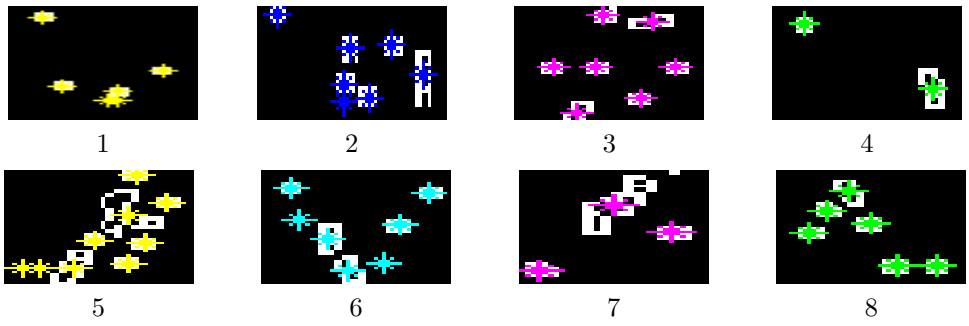


Figure 13: Inter-plane feature extraction from 8 clusters for the P1 plane using k-means clustering.

Algorithm 1 Training for video classification

INPUT: Gray color image of input image

A. For each gray image in training database:
   a. Apply canny edge operator on gray image.
   b. Apply unsupervised clustering on canny image to find clustering number.

B. Find the clustering number k_{inter} having more frequency using $\max_c \sum_{i=1}^{c} h_i$ from all the images, where i represents the unique cluster obtained from Step A, while h_i denotes the total contributed clusters in the i_{th} cluster.

OUTPUT: Cluster number (k_inter)

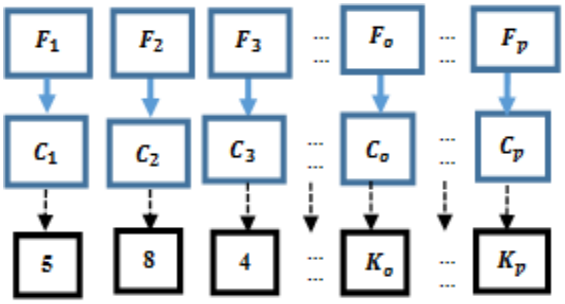
Algorithm 2 Testing for video classification

INPUT: Testing images in gray color and k_{Inter} obtained by Algorithm 1

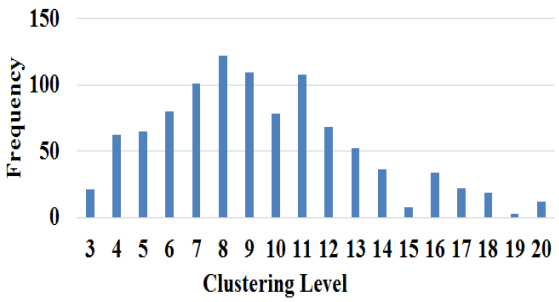
A. Apply Rough Fuzzy component grouping method on gray image to obtain 8 groups using the steps presented in Section 3.2 and shown in Fig. 8.

B. For each component group image shown in Fig. 8:
   a. For every pixel, calculate θ using $\tan^{-1}\frac{y}{x}$ to find angular planes, where all the pixels having the same angle belong to one angular plane.
   b. For each unique angular plane μ, apply k-means clustering, where the number of clusters has been set to k_{inter} obtained from the training Algorithm (1).
   c. For each clustering group l, across all the angular planes, and estimate the maximum frequency (v) of gradient component using $\max_{\mu,l}(\tan^{-1}\frac{y}{x})$ described in Section 3.3 (see Fig. 10).
   d. Compute covariance matrix (COV) using $\frac{1}{k_{inter}}\sum_{l=1}^{k_{inter}} v(v)^T$.
   e. Compute correlation matrix (COR) using $\frac{cov_i}{\sqrt{\frac{1}{k_{inter}}\sum_{l=1}^{k_{inter}} v(v)^T}}$.

OUTPUT: Covariance (COV) and correlation(COR) matrix.



(a) Cluster values of the k-means clustering with k=m for the 1000 sample frames.



(b) Histogram for choosing k value automatically.

Figure 14: Determination of k value for inter plane feature extraction.

3.5. Feature Extraction from Temporal Frames

Since the input video provides temporal frames, we exploit temporal information to increase the discriminative power of the feature extraction in this work. For the extracted features as discussed in the previous section, we extract the same the features for the left and right sides of the key frame if available. Otherwise, the proposed method considers three consecutive frames for feature extraction. It computes the average of the three feature matrices given by three frames, which gives the final feature matrix for classification. It is noted from literature that Neural Network (NN) is a nonlinear model and has the ability to identify complex nonlinear relationships between dependent and independent variables. As a result, it is a non-parametric model compared to parametric models that require higher statistical calculation. Although there are two other types of neural networks, namely, Radial Basis Function (RBF) networks and Learning Vector Quantization (LVQ) networks, feedforward perceptron trained with back propagation is used in solving problems for its higher degree of generalization from training data. It is noted that the feed forward neural network classifier used in [26] explores the above characteristics of NN for classifying text and non-text pixels in videos, we thus propose the neural network classifier in the same way for classification in this work [26].

Since the problem is 10 class classification, we consider 10 output nodes, one for each of the ten classes. Two intermediate layers are used in this classification. Every node on one layer is connected with the nodes on the previous layer. The output of a node is defined as a function of the weighted sum of the connected nodes in the previous layer. Here, neural network considers random values as the initial weights, and then updates the weights automatically during learning stage according to problems. For choosing training samples, we use a 10-fold cross validation procedure, which automatically provides the number of training

and testing samples for classification. In this work, we consider 30,000 frames for classification, including the temporal frames. Out of which 27,000 frames are used for training.

4. Experimental Results

We use YouTube and other internet sources for collecting the dataset for 10 classes, namely, Defense (D), Economics (Ec), Sports (S), Medical (M), Weather (W), Animation (A), e-learning (e-L), Technology (T), Outlet (O) and Animal Planet (AP), to evaluate the proposed classification method as there is no standard datasets available for the categorization of different scene type video in literature. We chose the above 10 classes as they play an important role in smart city and digital city development [1]. Each class consists of 3,000 frames, which includes three temporal frames for each keyframe. For the 10 classes, we get 30,000 frames for experimentation in this work. We believe the considered huge data are close to generalized data for the above mentioned 10 classes. As discussed in Introduction Section 1, each dataset poses different challenges, like low resolution, contrast, font, font size and background variations, multi-oriented texts, etc, due to different nature and characteristics. For example, the resolution range varies from 480$\times$ 360 to 1920$\times$ 1080.

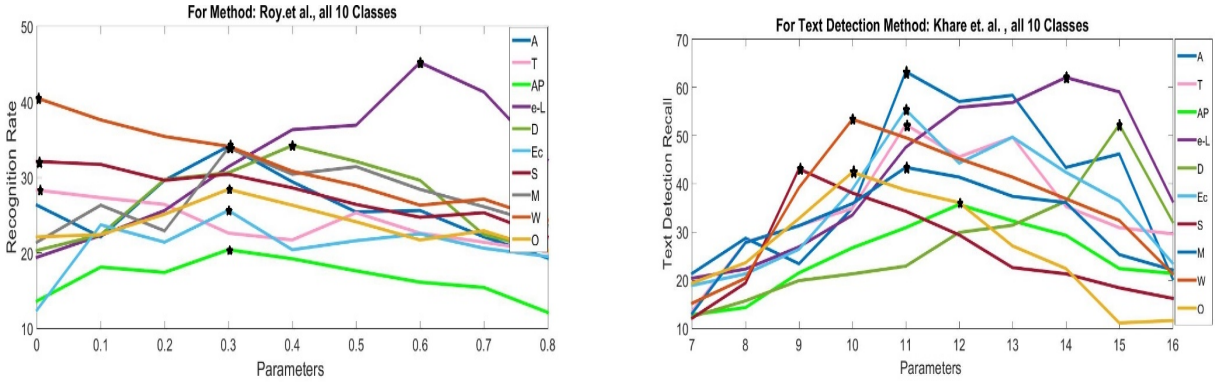
The proposed method involves three types of experiments to evaluate the performance of the proposed method: classification experiments, text detection experiments for validating the effectiveness of the classification step, and recognition experiment through binarization to show the usefulness of the proposed classification step. For evaluating the proposed classification step, we calculate standard classification rate through confusion matrices. For evaluating text detection results of different text detection methods, we follow the instructions given in [25–34], which use standard measures, namely, Recall (R), Precision (P) and F-measure (F). For recognition experiments, we calculate Recognition Rate (RR) at character level for different binarization methods. We also conduct experiments, namely, prior to classification and after classification for both text detection and recognition to show that the text detection and recognition method gives poor results for prior to classification and significant improvement after classification. Prior to classification considers frames of all the 10 classes as the input for experiments, while after classification considers individual classes as the input for experimentation. In general, after classification, text detection and recognition performance improves significantly because by considering the advantage of classification, the parameters are tuned in respective methods with the samples chosen randomly as discussed in Section 3.2.

To show the superiority of the proposed classification method, we implement the state of the art video classification methods as per the instructions given in these papers and use the same experimental set up as the proposed method for comparative studies, namely, Bosch et al.s method [14]which explores probabilistic latent semantic analysis and SIFT features for scene image classification, Dunlops method [15] which proposes scene classification of images and videos through semantic segmentation, and Qin et al.s method [20] which proposes statistical, structural and spatial features in color space with an SVM classifier for video text frame classification. The reason to choose the above three methods for comparative studies is that Bosch et al.s method focuses on scene image classification, Dunlops method focuses on video classification, which utilizes temporal frames like the proposed method, and Qin et al.s method focuses on video text frames as the proposed method.

To show the advantage of classification, we implement/use three types of text detection methods: the methods that use temporal frames for video text detection, namely, Khare et al. [28], Moselh et al.[33], Zhao et al. [32]and Li et al. [26], the methods that do not use temporal information, namely, Shivakumara et al. [27, 31], and the methods that are developed for text detection in natural scene images, namely, Yin et al. [6], Rong et al. [25] and Epshtein et al. [30]. We consider the methods which are developed for text detection in natural scene images for experimentation because the video contains scene texts in complex background, which share the same characteristics of text in natural scene images. In the similar way, we also consider three types of binarization methods for recognition experiments in this work: the method developed for text binarization in videos, namely, Roy et al. [35], the methods developed for binarizing text in natural scene images, namely, Milyaev et al. [37], and the methods developed for binarizing text in degraded document images, namely, Su et al. [36] and Howe [7]. The main reason to choose the three types of text detection and binarization methods is that as stated in the Introduction Section 1, the considered scene type video

classification is a complex problem which suffers from different challenges, such as contrast and background variations. As a result, the challenges influence directly on text in frames.

As discussed in Section 3 about the unified framework for text recognition as shown in Fig. 2, we present criteria for determining parameter values automatically here. For each class, the proposed method runs text detection and binarization methods by varying parameter values to calculate text detection rate and recognition rate, respectively. At some point, text detection or recognition rate reaches the highest score and starts decreasing as the parameter value changes. The value which reaches the highest score is considered as the actual parameter value for both text detection and recognition. For example, sample experiments to derive the parameter for window size defined in text detection method [28] is shown in Fig. 15a, where one can see lines graphs change as parameter values change according to classes. The peak points are marked by *star* for all the lines of classes. Similarly, sample experiments to derive threshold value used for binarization in [35] are shown in Fig. 15b, where lines graphs are changing as threshold values change according to classes.



(a) Parameter values for the window size defined in the text detection method [28].

(b) Parameter values for the threshold used for binarization in [35].

Figure 15: Sample experiments for deriving the parameter values of the text detection and binarization methods automatically using training samples for all 10 classes.

4.1. Evaluation of Classification Method

It is noted that the proposed method involves the following key steps for the classification of scene type videos, namely, covariance and correlation feature extraction for classified edge components by the combination of rough set and fuzzy, the use of intra and inter planes, and the use of Sobel edge images and Canny edge images. In order to analyze the contributions of the above key steps, we conduct experiments as follows, (1) covariance + intra + inter + classification, (2) correlation + intra + inter + classification, (3) covariance + correlation + intra + classification, (4) covariance + correlation + inter + classification, (5) covariance + correlation + intra + inter + classification using Sobel edges of the input frames, and (6) covariance + correlation + intra + inter + classification using Canny edge image of the input frame, which are required to analyze the contributions of covariance, correlation, intra plane features, inter plane features, Sobel edge detector and Canny edge detector, respectively. For each experiment, confusion matrices are respectively estimated as reported in Table 1-6 for the above 6 experiments. The average classification rate, which is the mean of diagonal elements of the confusion matrices for the respective 6 experiments are 55.6%, 58.9%, 54.0%, 63.0%, 56.3% and 76.0%. This shows that all the 6 key steps contribute almost equally except the features using inter planes and the proposed method with Canny edge detector. This indicates the features extracted for the edge components corresponding to clusters across planes have more discriminative power than intra planes. However, the proposed method with Canny edge detector and temporal information achieves the best average classification rate (76.0%) compared to the proposed method with Sobel edge detector and temporal information (56.3%). It is true that Sobel edge detector is good for high contrast images as it involves the first order derivative with one optimal threshold, while Canny edge

Table 1: Confusion matrix of covariance+intra+inter+classification (Average of diagonal element is 55.6%).

Class	D	Ec	S	M	W	A	E-l	T	En	AP
D	**48.1**	9	3	11.3	4.1	2.2	3.2	4.1	4.7	10.3
Ec	4.6	**56.7**	4.2	3.5	7.3	2.6	6.1	6.3	3.6	5.1
S	10.4	1.1	**50.6**	4.1	7.2	4.9	1.8	13.3	3.9	2.7
M	2.1	3.7	4.2	**52.2**	1.1	15.2	0.7	2.3	5.6	12.9
W	4.1	2.9	1.4	3.8	**55.8**	18.8	7.7	2.9	0	2.6
A	11.9	3.7	8.9	4.8	1.2	**50.1**	5.1	0	0	14.3
E-l	2.1	0	3.7	6.2	12.8	3.5	**57.8**	5.2	5.1	3.6
T	1.4	19.2	2.6	4.8	3.9	2.8	0.8	**54.9**	6.1	3.5
En	16.2	3.8	2.4	1.3	0	2.1	4.1	4.8	**65.3**	0
AP	3.6	5.8	2.8	4.9	5.3	4.3	2.7	2.4	4.1	**64.1**

Table 2: Confusion matrix of correlation+intra+inter+classification (Average of diagonal element is 58.92%).

Class	D	Ec	S	M	W	A	e-L	T	O	AP
D	**40.3**	1.2	11.4	2.8	1.9	4.1	4.8	13.3	3.9	16.3
Ec	9.1	**63.8**	0.9	1.2	2.3	6.7	8.1	1.2	4.6	2.1
S	4.2	2.8	**66.2**	1.1	3.1	1.4	10.8	5.9	2.1	2.4
M	7.3	1.6	3.3	**69.2**	3.7	4.1	5.2	2.3	2	1.3
W	8.4	3.9	7.1	1.8	**61.4**	4	1.3	4.7	7.4	1
A	5.8	1.8	3.4	11.3	1.2	**56.7**	4.9	0.4	1.9	12.6
e-L	5.2	2.1	19.1	0.2	1.1	1.4	**54.7**	12.4	2.8	1
T	0	12.3	1.5	10.4	2.8	15.3	2.1	**50.6**	3.3	1.7
O	2.5	4.1	7.1	3.1	1.9	3.9	6.1	5.5	**64.5**	1.3
AP	11.7	1.2	0.6	1.3	0	2.7	3.1	5.7	11.9	**61.8**

detector is good for both low and high contrast images as it involves double optimal thresholds. At the same time, the considered dataset contains images of different contrasts variations. Therefore, the proposed method with Canny edge detector and temporal information scores better results compared to the proposed method with Sobel edge detector and temporal information. There is a significant difference when we compare the average classification rates of covariance, correlation, intra and inter with the proposed method (76.0%). This is due to the integration of strengths of covariance-correlation among pixels in intra and inter angular planes given by rough-fuzzy combination, which extracts unique stroke distributions locally and globally from irregular edge patterns in edge components of frames, and temporal information which adds stability to features by considering information in neighboring frames. Therefore, we can conclude that the proposed method aggregates the advantages of a new way of combination of rough-fuzzy for grouping edge components, covariance-correlation of intra, inter planes, Canny edge detector, and temporal information of video to achieve better results for the complex classification problem.

Samples of successful classification results of the proposed method are shown in Fig. 16, where one can see that the proposed method classifies different scene type video frames correctly. It is noted from literature review on classification that deep learning using convolutional network is popular because it has the ability to solve complex problems with better performance. It is also noted from the literature review [23, 24], that a deep learning framework has the following limitations which affect to develop a generalized classification system for diversified video. They require a large number of training samples, large computational resources (GPU), high time complexity for learning models, optimization algorithms to adjust network parameters, the implementation of deep learning algorithms on mobile devices is not simple, the analysis of the stability of deep neural networks is hard, deep neural network for non-linear networked control systems is still an issue, etc.

To validate the above analysis, we conduct experiments and compared with the proposed method on

Table 3: Confusion matrix of covariance+correlation+intra+classification (Average of diagonal element is 54.09%).

Class	D	Ec	S	M	W	A	E-I	T	En	AP
D	**49.8**	3.4	1.3	2	2.5	3.2	13.4	8.3	5.9	10.2
Ec	3.5	**42.1**	3.8	2.9	14.8	10.3	3.9	12.1	2.8	3.8
S	7.1	2.8	**56.7**	4.5	3.9	4.6	3.9	9.8	3.1	3.6
M	3.6	3.1	8.2	**57.1**	3.4	2.8	9.8	4.8	2.3	4.9
W	2.8	1.3	3.8	12.9	**59.1**	8.5	2.4	1.7	1.7	5.8
A	6.2	1.8	2.8	2.4	4.5	**51.2**	6.9	1.4	10.2	12.6
E-l	14.9	4.8	2.5	0	1.3	1.7	**57.8**	2.8	2.8	11.4
T	10.8	6.8	0	1.2	3.2	2.7	2.9	**49.8**	19.7	2.9
En	0.5	12.5	1.8	4.6	1.2	11.9	3.1	1.4	**56.2**	6.8
AP	10.9	1.8	2.1	2.5	0.7	3.1	3	3.5	11.3	**61.1**

Table 4: Confusion matrix of covariance+correlation+inter+classification (Average of diagonal element is 63.08%).

Class	D	Ec	S	M	W	A	e-L	T	O	AP
D	**60.1**	10.3	1.4	2.1	12.6	2.6	1.8	2.1	2.9	4.1
Ec	4.7	**66.7**	4.7	2.9	1.2	3.3	12	0.2	2.1	2.2
S	0	2.9	**69.6**	1.3	8.4	1.1	6.2	0.2	7.1	3.2
M	3.7	2.8	3.9	**65.5**	2.1	2.8	1.9	0.4	10.8	6.1
W	3.1	4.9	4.3	1.6	**59.8**	0.7	0.4	12.8	5.2	7.2
A	10.1	1.9	3.5	1.7	0.6	**62.2**	2.8	3.9	10.8	2.5
e-L	3.7	2.8	2.7	0.8	1.8	14.3	**56.3**	7.2	0.4	10
T	10	4.4	1.8	1.3	2.1	1.7	3.1	**64.6**	8.9	2.1
O	7.7	3.9	1.7	2.9	9.1	1.1	0.5	3.1	**67.4**	2.6
AP	13.1	2.6	3.1	13.3	0.5	2.1	3.1	2.4	1.2	**58.6**

Table 5: Confusion matrix of the proposed method using Sobel edge components with temporal frames (Average of diagonal element is 56.3%).

Class	D	Ec	S	M	W	A	e-L	T	O	AP
D	**61.2**	2.3	1.2	3.8	11.2	6.3	5.1	4.6	3.1	1.0
Ec	5.3	**58.7**	1.3	2.3	1.9	4.8	2.9	1.0	7.3	14.2
S	8.0	1.0	**52.9**	16.0	5.0	2.1	3.3	4.8	2.5	4.0
M	2.5	3.0	5.0	**67.0**	2.4	3.9	0.3	1.9	11.8	2.9
W	6.3	2.0	8.2	3.0	**49.1**	2.7	4.0	12.7	6.0	5.5
A	2.1	10.0	15.7	1.2	12.9	**44.2**	5.3	3.7	2.0	2.5
e-L	5.2	2.4	6.0	4.3	3.0	7.5	**55.0**	12.0	2.3	1.9
T	11.0	1.2	2.0	7.6	13.0	2.6	5.2	**50.0**	1.8	5.2
O	4.1	3.2	2.4	6.8	8.0	4.2	1.0	6.0	**63.3**	0.7
AP	2.1	1.3	5.2	13.0	2.4	3.2	1.9	1.0	7.5	**62.0**

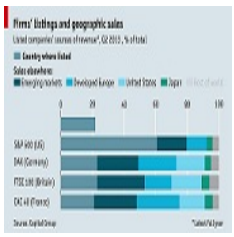
Table 6: Confusion matrix of the proposed method using Canny edge components with temporal frames (Average of diagonal element is 76.0%).

Class	D	Ec	S	M	W	A	e-L	T	O	AP
D	**83.9**	1.03	2.7	1.7	2.23	2.3	1.07	1.6	2.4	1.04
Ec	1.3	**75.5**	2.04	1.04	1.7	2.67	4.22	1.53	5.8	4.2
S	2.21	6.47	**72.7**	2.8	4.7	1.05	3.9	2.4	1.05	2.69
M	5.8	1.36	2.8	**71.5**	4.06	6.53	1.23	2.12	3.07	1.49
W	1.8	2.3	1.32	1.17	**80.6**	3.5	1.47	2.53	3.71	1.56
A	1.4	3.26	2.51	2.4	1.1	**77.7**	2.78	4.9	2.41	1.54
e-L	1.4	3.26	2.51	2.4	1.1	1.78	**78.7**	4.9	2.41	1.54
T	1.83	3.28	6.9	1.07	1.82	1.32	2	**75.8**	1.78	4.2
O	12.9	1.82	2.96	1.06	1.93	2.45	1.48	3.73	**69.5**	2.17
AP	1.37	2.78	1.05	2.67	6.58	4.72	2.47	6.48	2.38	**69.5**

classification using GOOGLE API [16], which is available publicly and uses deep learning, cloud, and a large number of features for retrieving scene images that contain multiple objects in each image. The purpose of doing experiments with this system is to show that the performance of the systems which involve deep learning that depends heavily on a number of labeled samples and setting optimal parameters to achieve good results. Besides, this set up may not be feasible for the data which consists of a small number of samples, or when background complexity varies greatly for samples of the same class. We generate confidence scores for training samples of our data using GOOGLE API. This process gives different labels for each class with confidence scores. For instance, Animal Planet (AP) class can have agriculture, black bird, branch, nature labels, etc, while Animation can have amusement park, amusement ride, atmosphere, etc. Labels are given by GOOGLE API system. In this way, we create feature vectors for all the 10 classes based on training samples. We set 85% as a cut off threshold to confidence score to generate the final confusion matrix for all the 10 classes. This 85% cut of is fixed based on the experiments on training samples. It is observed from experiments that if we increase the cut off value, the method includes irrelevant labels and if we decrease, it loses relevant labels.

The quantitative results of the proposed method and GOOGLE API are reported in Table 6 and Table 7. According to our analysis, it is noted that the GOOGLE API system works when it recognizes multiple objects correctly in images. If the image contains an object which is not trained by the system, GOOGLE API fails to classify the image correctly, while the proposed method is not trained on specific shapes of objects, rather it studies the pattern of edge components using Fuzzy and rough set combination, thus it gives better results for our dataset. However, if we train GOOGLE API with our dataset, it may score better results than the proposed method. But this is the limitation of the GOOGLE API system as its performance depends on the number of labeled samples. On the other hand, the proposed method does not require such large number of samples for achieving good results. In addition, according to website [16] and experiments, it is noticed that GOOGLE API works based on shapes of multiple objects in scene images. However, in case of our dataset, one cannot expect particular shapes of objects because scene type images of our dataset may contain objects or may not. For example, Whether and Economic scene classes do not contain any object with particular shapes. Therefore, GOOGLE API scores poor results compared to the proposed method. The quantitative results of Bosch et al.s method [14], Dunlops method [15] and Qin et al.s method [20] are reported in Table 8-Table 10, respectively. It is observed from Table 8-Table 10 that the proposed method is better than the existing methods. The reason for the poor results of the existing methods is that they require multiple objects to train classifiers. On the other hand, the proposed method extracts unique shapes from irregular edge patterns by exploring rough-fuzzy and distinct relationship among pixels locally and globally based on covariance-correlation of intra, inter planes and temporal information. Therefore, the proposed method is the best compared to the existing classification methods.

Defense



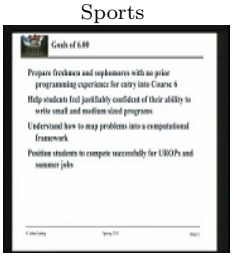
Economics

Sports

Medical

Weather

Animation



e-Learning

Technology

Outlet

Animal Plant

Figure 16: Samples of successful classification results of the proposed method.

Table 7: Confusion matrix of GOOGLE API system [16].

Class	**D**	**Ec**	**S**	**M**	**W**	**A**	**e-L**	**T**	**O**	**AP**
D	**72.8**	3.1	2.4	3.1	2.9	3.2	2.9	2.4	4.7	2.4
Ec	3.0	**73.1**	2.4	2.5	2,5	2.7	6.9	2.1	2.1	2.6
S	2.5	2.4	**78.6**	2.3	2.4	2.2	2.5	2.1	2.9	2.1
M	2.7	2.0	2.7	**77.4**	2.9	2.6	2.6	2.6	2.1	2.5
W	2.5	2.5	2.2	4.2	**72.8**	3.6	3.2	2.4	3.8	2.7
A	2.5	2.8	2.1	2.5	3.3	**68.7**	3.9	3.5	5.7	5.0
e-L	2.5	3.7	2.0	2.5	2.1	2.6	**75.6**	3.2	3.0	2.7
T	3.1	2.2	2.1	2.9	2.9	4.4	3.4	**72.4**	4.3	2.3
O	5.5	3.9	4.5	3.7	6.6	5.8	8.2	9.5	**48.9**	3.4
AP	2.6	2.7	2.5	2.3	2.1	3.2	2.3	2.7	2.8	**76.8**

Table 8: Confusion matrix of the Bosch et al. [14] classification.

Class	**D**	**Ec**	**S**	**M**	**W**	**A**	**e-L**	**T**	**O**	**AP**
D	**60.4**	3.7	1.2	12.0	2.9	6.0	1.2	4.0	1.3	7.3
Ec	0.5	**90.5**	0.0	0.5	0.3	2.4	5.2	0.6	0.0	0.0
S	0.5	1.0	**94.4**	2.2	0.9	0.5	0.0	0.2	0.2	0.0
M	1.1	1.8	0.2	**90.3**	1.7	0.8	0.3	0.4	0.4	3.0
W	0.3	0.6	1.4	5.0	**81.8**	2.1	1.0	0.3	0.4	7.1
A	5.5	5.1	2.3	8.3	2.8	**49.2**	3.1	5.4	7.8	10.4
e-L	3.1	13.7	1.7	2.5	0.2	4.0	**71.8**	1.0	1.0	1.0
T	24.4	10.4	4.6	4.9	3.4	7.4	2.5	**32.8**	7.2	2.5
O	5.6	1.6	5.6	12.6	5.3	13.9	0.7	13.3	**32.8**	8.6
AP	18.9	0.6	0.7	1.0	10.6	2.1	0.4	0.3	1.0	**64.5**

4.2. Validating Classification Through Text Detection Methods

As mentioned in Section 4, to show the advantage of the proposed classification, we propose to calculate text detection and recognition rates of different text detection and binarization methods prior to classification (which considers all the frames for calculating text detection rates and texts of all the frames for calculating recognition rates) and after classification (frames of individual class classified by the proposed and the existing classification methods as input for calculating text detection and recognition rates). To know the effect of the proposed and the existing classification methods in terms of text detection and recognition performance after classification, we calculate the averages for Recall (R), Precision (P), F-Measure (F) and

Table 9: Confusion matrix of the Dunlop [15] classification.

Class	**D**	**Ec**	**S**	**M**	**W**	**A**	**e-L**	**T**	**O**	**AP**
D	**71.4**	5.1	1.0	7.5	4.7	2.1	1.3	1.7	0.8	4.5
Ec	2.1	**87.9**	0.8	1.2	1.0	2.6	2.0	0.9	0.5	0.9
S	1.0	1.1	**90.1**	3.1	0.7	0.7	1.2	0.6	1.0	0.6
M	1.4	1.6	1.3	**86.3**	3.3	1.1	1.4	1.5	0.9	1.0
W	1.2	1.4	1.0	3.9	**84.2**	0.9	2.6	1.4	0.6	2.9
A	6.5	8.1	7.4	9.1	5.8	**46.6**	4.1	3.6	3.8	5.1
e-L	3.5	10.5	1.8	1.4	0.5	2.6	**77.1**	0.9	0.7	0.9
T	15.3	1.7	7.7	5.4	7.1	1.3	4.4	**54.0**	1.9	1.3
O	9.5	2.2	11.4	14.3	6.3	2.2	1.6	10.1	**39.0**	3.6
AP	6.8	1.1	2.3	2.8	13.7	0.5	2.9	0.9	0.6	**68.6**

Table 10: Confusion matrix of the Qin et al.[20] classification.

Class	D	Ec	S	M	W	A	e-L	T	O	AP
D	**53.1**	15.3	1.8	7.4	8.8	1.1	2.7	0.5	0.3	9.0
Ec	0.0	**97.9**	0.2	0.0	0.1	1.3	0.0	0.0	0.0	0.5
S	1.5	0.3	**85.6**	0.5	4.3	2.3	1.9	2.7	0.4	0.4
M	2.3	1.7	1.3	**80.1**	3.3	2.7	4.2	0.1	0.1	4.2
W	0.9	0.8	0.8	2.1	**93.6**	0.0	0.0	0.1	1.1	0.6
A	0.6	13.8	2.8	0.5	2.2	**74.7**	2.2	0.6	0.5	1.9
e-L	2.8	0.5	3.0	2.9	2.9	1.5	**80.5**	1.8	0.7	3.4
T	6.8	10.1	13.1	14.9	4.4	2.2	2.8	**37.5**	2.6	5.6
O	5.6	6.7	13.7	11.8	7.2	14.7	13.7	7.4	**7.7**	11.6
AP	1.1	1.0	1.7	1.7	1.8	0.7	0.1	0.1	0.0	**91.7**

Recognition Rate (RR) of classes for respective classification methods including the proposed classification method. Therefore, we report average recall, precision and F-measure of the different text detection methods for each classification methods on the 10 classes in Table 11. For experiments prior to classification, we use default parameters used in the text detection methods to calculate recall, precision and F-measure. However, for the experiments after classification, since classes are known by the classification methods, we tune key parameters of text detection methods based on training samples of each class to calculate recall, precision and F-measures according to the complexity of the classes. We determine the parameters, namely, window size, aspect ratio, window size, aspect ratio for stroke width, threshold for Bayesian classifier outputs, window size, threshold for features vector, aspect ratio for stroke width and the number of sub-blocks for the text detection methods listed in Table 11, respectively. It is observed from Table 11 that the recall, precision and F-measure of all the text detection methods improve significantly compared to the recall, precision and F-measure prior to classification. This shows that classification is useful for enhancing the performances of text detection methods especially when we have frames with large variations in background and foreground complexities. At the same time, when we compare text detection performance for the proposed classification with the existing classification methods, most of the text detection methods score highest F-measure for the proposed classification method compared to the existing classification methods. We believe that if the classification methods classify frames correctly without many misclassification errors, text detection methods score good results. Therefore, since the proposed classification achieves the best classification rate compared to the existing classification methods as discussed in Section 4.1, most of the text detection methods perform better for the proposed classification compared to the existing classification methods. However, Li et al.'s [26] method scores the best F-measure for Bosch et al.'s [14] classification, Rong et al.'s [25] method scores the best F-measure for Dunlop's [15] classification, Shivakumara et al.'s method [31] scores the best F-measure for Qin et al.'s [20] classification, and Zhao et al.'s [32] method scores the best F-measure for GOOGLE API classification [16].

4.3. Effectiveness of Classification Through Binarization Methods

In the same way of text detection experiment for each classification methods as discussed in the previous section, we report average Recognition Rate (RR) of the different binarization methods for the classes of respective classification methods in Table 12. We determine key parameters from each binarization method listed in Table 12, such as threshold value for Bayesian classifier [35], threshold value for Canny edge image [7], and window sizes for different binarization algorithms [37]. Since Su et al. [36] provided exe files that find values automatically, there is no option for tuning. It can be seen from Table 12 that the RR after classification improve greatly for all the binarization methods of all the classification methods compared to the RR of prior to classification. This is possible because of tuning the parameters based on samples of each class by considering advantage of classification step. When we compare the recognition performance of the proposed classification with the existing classification, most binarization methods achieve the best recognition rate for the proposed classification compared to the existing classification. However, Milyaev et

Table 11: Text detection performance of the different existing methods prior to classification and after classification for proposed and existing classification methods on data of 10 classes. PC denotes "Prior to classification" and AC denotes "After classification".

Text Detection Methods		**PC**	**AC**				
			Proposed	**[14]**	**[15]**	**[20]**	**[16]**
[28]	**R**	32.2	50.3	43.3	50.1	53.3	40.3
	P	40.5	55.7	53.2	53.2	47.6	59.7
	F	35.8	52.6	47.7	51.6	50.2	48.1
[6]	**R**	42.3	55	52.6	45.1	47.4	50.4
	P	48.4	58.6	46.8	47.7	42.4	53.5
	F	45.1	56.5	49.5	46.3	44.7	51.9
[25]	**R**	33.2	45.5	51.3	55.1	44.1	42.5
	P	37.2	55.6	50.4	52.9	50.4	57.2
	F	35	50	50.8	52.9	47	48.7
[33]	**R**	35.4	52	47.2	56.1	52.8	54.2
	P	44.1	56.8	56.2	46.1	47.2	50.2
	F	39.2	53.7	51.3	50.6	49.8	52.1
[31]	R	33.6	51.3	55.1	55.8	52.5	53.2
	P	42.7	51.4	49.9	52.8	58.4	45.4
	F	37.6	50.4	52.3	54.2	55.2	48.9
[32]	**R**	32.6	45.5	40.2	42.9	47.1	43.4
	P	39.6	52.6	47.3	48.8	52.3	53.5
	F	35.7	47.8	43.4	45.6	59.5	53.5
[27]	**R**	38.4	50.9	51.8	46.9	43.3	53.2
	P	27.2	53.1	38.9	55.2	50.1	40.3
	F	31.8	51.6	44.4	50.7	46.5	45.8
[30]	**R**	31.6	53.1	54.7	50.2	46.2	50.2
	P	35.8	59	53.3	52.8	45.8	57
	F	33.8	55.7	53.9	51.4	45.9	53.3
[26]	**R**	31.3	41.9	46.2	40.1	45.7	42.8
	P	37.4	53	54.9	56.3	48.2	51.2
	F	34	46.4	50.1	46.8	46.9	46.6

al.s method [37] scores the best recognition rate for GOOGLE API classification. The reason for the poor recognition performances by different binarization methods for the existing classification is the same as the reason discussed in the previous section. On the other hand, since the proposed classification method is better than the existing classification in terms of classification rate as discussed in Section 4.1, binarization methods perform better compared to the existing classification methods.

In summary, we can assert that classification of frames of different complexity helps in enhancing the performance of text detection and recognition methods. In addition, as classification rate of the classification methods increases, one can expect better text detection and recognition performance by different text detection and binarization methods. In order to test the generic nature and robustness of the proposed classification in terms of text detection and recognition, we choose 5 new classes, namely, Recipes of Cooking (RC) which contains text as Animal Planet, Craft Making (CM) which contains caption text as Animal Planet, Indian Classical Musical Concert (ICMC) which contains texts as in Sports, Outlet, Defense, Teleshopping (TS) which contains caption texts as in Sports, Animal Planet, and Yoga (Y) which contain caption texts as in Animal Planet. The sample images of new classes with texts are shown in Fig. 17.

For experimentation, we use the same setup that we have used for the 10 classes database to calculate different measures for text detection and binarization methods. The quantitative results of different text detection and binarization methods for the proposed and existing classification on the data of 5 new classes

Recipes Cooking | Teleshopping | Yoga

Craft Making | Indian Classical Music Concert

Figure 17: Samples of video frames of new 5 classes with text detection by [6].

are reported in Table 13 and Table 14, respectively. With the same parameter setup, the results reported in Table 13 and Table 14 show that the text detection and recognition performance of the text detection and binarization methods improves significantly after classification compared those of prior to classification with the similar conclusion we have drawn for the data of 10 classes. In the same way, most of the text detection methods and binarization methods give better results for the proposed classification compared to existing classification. Furthermore, it is noted from the results of 10 classes and 5 new classes, the detection and recognition rates report almost similar patterns after classification. In summary, from the above experimental analysis, one can confirm that the proposed classification has the ability to extend to a number of new classes, and the performance of classification is independent from the content of frames in terms of text detection and recognition rates. Hence, the proposed method is generic and consistent to different classes or different contents of frames. This is because of the use of flexible rough-fuzzy combination, covariance-correlation for intra, inter planes and temporal information. Since our aim is to show the effectiveness of the classification method, we tune parameters of the text detection and binarization methods. As a result, the improved results after classification is not high as plain document analysis accuracy which usually has more than 90% accuracy. However, this work shows direction that one can modify the existing methods or develop new methods according to the complexity of individual classes for achieving still better performance of text detection and recognition by considering the advantage of classification.

Table 12: Average recognition rate (%) of the different binarization methods for the proposed and existing classification methods on data of 10 classes.

Methods	**[35]**	**[36]**	**[7]**	**[37]**
Prior to classification	15.78	12.21	13.14	10.31
Classification methods	**Before and after classification**			
Proposed	**32.3**	**40.1**	**37.1**	37.2
[14]	23.2	28.2	24.7	35.8
[15]	28.9	23	22.2	30.4
[20]	30.9	27	20.2	19.4
[16]	31.8	36.7	35.2	**39.3**

Table 13: Text detection performance of the different existing methods prior to classification and after classification for proposed and existing classification methods on new 5 classes. PC denotes "Prior to classification" and AC denotes "After classification".

Text Detection Method		**PC**	**AC**				
			Proposed	**[14]**	**[15]**	**[20]**	**[16]**
[28]	**R**	34.7	48.7	45.2	54.1	51.6	42.1
	P	45.4	54	52.5	50.9	46.2	45.4
	F	40	51.3	48.5	52.4	48.7	43.6
[6]	**R**	46.1	54	50.8	52.6	35.4	51.2
	P	50.2	56.5	42.5	47.4	39.8	49.1
	F	48.1	55.2	46.2	47	38.6	50.1
[25]	**R**	35.8	45.2	43.5	44.1	47.3	47.2
	P	38.9	50.9	51.8	41.5	48.5	54.1
	F	37.3	48.04	42.9	43.2	44.7	50.4
[33]	**R**	37.9	54.9	47.3	50.3	53.1	52.1
	P	48.2	58.8	48.5	53.1	45.4	57.5
	F	43	56.9	44.7	46	47.2	54.6
[31]	**R**	38.7	55.9	50.4	54.3	57.2	47.6
	P	42.3	59.5	57.1	57.1	44.5	54.2
	F	40.2	57.7	46.1	47.6	48.7	50.6
[32]	**R**	39.7	59.9	41.5	44.1	48.6	47.2
	P	42.8	55	43.8	50.2	44.1	57.8
	F	41.2	52.4	41.9	43.2	45.3	51.9
[27]	**R**	42.7	53.2	52.5	50.1	52.5	56.1
	P	35.3	56.4	48.4	44.5	48.1	52.2
	F	39	54.8	46.9	45.9	46.9	54.1
[30]	**R**	37.4	49.1	50.2	48.5	51.5	52.6
	P	39.9	54.4	43.4	47.5	38.2	54.2
	F	38.6	51.7	46	45.3	46.5	53.3
[26]	**R**	36.1	49.2	48.3	41.7	48.7	42.4
	P	39.4	51.4	41.1	42.8	44.5	50.5
	F	37.7	49.9	45.2	42	45.3	46

Table 14: Average recognition rate (%) of the different binarization methods for the proposed and existing classification methods on data of 5 new classes.

Methods	**[35]**	**[36]**	**[7]**	**[37]**
Prior to classification	19.7	16.7	18.3	17.9
Classification methods	**Before and after classification**			
Proposed	**35.1**	**33.1**	**27.8**	**36.6**
[14]	23.1	34.7	21.4	32.9
[15]	24.4	22.1	24.5	23.1
[20]	20.8	23.4	21.2	30.4
[16]	30.5	33.1	27.8	32.9

5. Conclusion and Future Work

We have proposed a novel method for scene type video categorization of different classes by exploring rough set and fuzzy logic combination. The combination classifies edge components in each input frame into different groups to extract local information. For each group, the proposed method extracts covariance and correlation features for intra and inter planes, which helps us encode unique relationship for each video class type. Temporal information is used to increase the discriminative power of feature extraction. The extracted features are then fed to a neural network classifier for the final classification. Experimental results on classification show that the proposed method works well for different scene type videos compared to the existing state of the art methods. In addition, the usefulness and effectiveness of the proposed classification is validated by text detection and recognition experiments with several other methods. However, it is noticed from experimental results that the text detection and recognition results decreases when misclassification occurs. Therefore, we have a plan to investigate and introduce an unsupervised method to determine the number of classes in the near future.

Acknowledgement

The work described in this paper was supported by the National Natural Science Foundation of China under Grant No. 61672273, No. 61272218 and No. 61321491, the Science Foundation for Distinguished Young Scholars of Jiangsu under Grant No. BK20160021, and partly supported by the University of Malaya HIR under Grant No: M.C/625/1/HIR/210.

References

[1] M. M. Rathore, A. Ahmad, A. Paul, S. Rho, Urban planning and building smart cities based on the internet of things using big data analytics, Computer Networks 101 (2016) 63–80.

[2] R. Minetto, N. Thome, M. Cord, N. J. Leite, J. Stolfi, Snoopertext: A text detection system for automatic indexing of urban scenes, Computer Vision and Image Understanding 122 (2014) 92–104.

[3] G. Zhu, C. Xu, Q. Huang, Y. Rui, S. Jiang, W. Gao, H. Yao, Event tactic analysis based on broadcast sports video, IEEE Transactions on Multimedia 11 (1) (2009) 49–67.

[4] J. R. Uijlings, A. W. Smeulders, R. J. Scha, Real-time visual concept classification, IEEE Transactions on Multimedia 12 (7) (2010) 665–681.

[5] Q. Ye, D. Doermann, Text detection and recognition in imagery: A survey, IEEE transactions on pattern analysis and machine intelligence 37 (7) (2015) 1480–1500.

[6] X.-C. Yin, X. Yin, K. Huang, H.-W. Hao, Robust text detection in natural scene images, IEEE transactions on pattern analysis and machine intelligence 36 (5) (2014) 970–983.

[7] N. R. Howe, Document binarization with automatic parameter tuning, International Journal on Document Analysis and Recognition (IJDAR) 16 (3) (2013) 247–258.

[8] Tesseract. http://code.google.com/p/tesseract-ocr/.

[9] N. Sharma, R. Mandal, R. Sharma, P. P. Roy, U. Pal, M. Blumenstein, Multi-lingual text recognition from video frames, in: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, IEEE, 2015, pp. 951–955.

[10] Z. Shou, D. Wang, S.-F. Chang, Temporal action localization in untrimmed videos via multi-stage cnns, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1049–1058.

[11] R. Ewerth, K. Ballafkir, M. Muhling, D. Seiler, B. Freisleben, Long-term incremental web-supervised learning of visual concepts via random savannas, IEEE Transactions on Multimedia 14 (4) (2012) 1008–1020.
[12] Y.-Y. Chen, W. H. Hsu, H.-Y. M. Liao, Automatic training image acquisition and effective feature selection from community-contributed photos for facial attribute detection, IEEE Transactions on Multimedia 15 (6) (2013) 1388–1399.
[13] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei-Fei, Large-scale video classification with convolutional neural networks, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2014, pp. 1725–1732.
[14] A. Bosch, A. Zisserman, X. Muñoz, Scene classification using a hybrid generative/discriminative approach, IEEE transactions on pattern analysis and machine intelligence 30 (4) (2008) 712–727.
[15] H. Dunlop, Scene classification of images and video via semantic segmentation, in: Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on, IEEE, 2010, pp. 72–79.
[16] https://cloud.google.com/vision/.
[17] J. Liu, C. Chen, Y. Zhu, W. Liu, D. N. Metaxas, Video classification via weakly supervised sequence modeling, Computer Vision and Image Understanding 152 (2016) 79–87.
[18] D. Tian, H. Sun, A. Vetro, Keypoint trajectory coding on compact descriptor for video analysis, in: Image Processing (ICIP), 2016 IEEE International Conference on, IEEE, 2016, pp. 171–175.
[19] Z. Xu, J. Hu, W. Deng, Recurrent convolutional neural network for video classification, in: Multimedia and Expo (ICME), 2016 IEEE International Conference on, IEEE, 2016, pp. 1–6.
[20] L. Qin, P. Shivakumara, T. Lu, U. Pal, C. L. Tan, Video scene text frames categorization for text detection and recognition, in: Pattern Recognition (ICPR), 2016 23rd International Conference on, IEEE, 2016, pp. 3886–3891.
[21] K. Nogueira, O. A. Penatti, J. A. dos Santos, Towards better exploiting convolutional neural networks for remote sensing scene classification, Pattern Recognition 61 (2017) 539–556.
[22] E. Ohn-Bar, M. M. Trivedi, Are all objects equal? deep spatio-temporal importance prediction in driving videos, Pattern Recognition 64 (2017) 425–436.
[23] A. Sharma, et al., Adapting off-the-shelf cnns for word spotting & recognition, in: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, IEEE, 2015, pp. 986–990.
[24] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F. E. Alsaadi, A survey of deep neural network architectures and their applications, Neurocomputing 234 (2017) 11–26.
[25] L. Rong, W. Suyu, Z. Shi, A two level algorithm for text detection in natural scene images, in: Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on, IEEE, 2014, pp. 329–333.
[26] H. Li, D. Doermann, O. Kia, Automatic text detection and tracking in digital video, IEEE transactions on image processing 9 (1) (2000) 147–156.
[27] P. Shivakumara, T. Q. Phan, C. L. Tan, New fourier-statistical features in rgb space for video text detection, IEEE transactions on circuits and systems for video technology 20 (11) (2010) 1520–1532.
[28] V. Khare, P. Shivakumara, P. Raveendran, A new histogram oriented moments descriptor for multi-oriented moving text detection in video, Expert Systems with Applications 42 (21) (2015) 7627–7640.
[29] G. Liang, P. Shivakumara, T. Lu, C. L. Tan, Multi-spectral fusion based approach for arbitrarily oriented scene text detection in video images, IEEE Transactions on Image Processing 24 (11) (2015) 4488–4501.
[30] B. Epshtein, E. Ofek, Y. Wexler, Detecting text in natural scenes with stroke width transform, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 2963–2970.
[31] P. Shivakumara, R. P. Sreedhar, T. Q. Phan, S. Lu, C. L. Tan, Multioriented video scene text detection through bayesian classification and boundary growing, IEEE Transactions on Circuits and systems for Video Technology 22 (8) (2012) 1227–1235.
[32] X. Zhao, K.-H. Lin, Y. Fu, Y. Hu, Y. Liu, T. S. Huang, Text from corners: A novel approach to detect text and caption in videos, IEEE Transactions on Image Processing 20 (3) (2011) 790–799.
[33] A. Mosleh, N. Bouguila, A. B. Hamza, Automatic inpainting scheme for video text detection and removal, IEEE Transactions on image processing 22 (11) (2013) 4460–4472.
[34] L. Wu, P. Shivakumara, T. Lu, C. L. Tan, A new technique for multi-oriented scene text line detection and tracking in video, IEEE Transactions on Multimedia 17 (8) (2015) 1137–1152.
[35] S. Roy, P. Shivakumara, P. P. Roy, U. Pal, C. L. Tan, T. Lu, Bayesian classifier for multi-oriented video text recognition system, Expert Systems with Applications 42 (13) (2015) 5554–5566.
[36] B. Su, S. Lu, C. L. Tan, Robust document image binarization technique for degraded document images, IEEE transactions on image processing 22 (4) (2013) 1408–1417.
[37] S. Milyaev, O. Barinova, T. Novikova, P. Kohli, V. Lempitsky, Image binarization for end-to-end text understanding in natural images, in: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on, IEEE, 2013, pp. 128–132.
[38] Y. Wei, Z. Zhang, W. Shen, D. Zeng, M. Fang, S. Zhou, Text detection in scene images based on exhaustive segmentation, Signal Processing: Image Communication 50 (2017) 1–8.
[39] Y. Zhang, W. Wang, L. Wang, L. Wang, Scene text recognition with deeper convolutional neural networks, in: Image Processing (ICIP), 2015 IEEE International Conference on, IEEE, 2015, pp. 2384–2388.
[40] S. Yousfi, S.-A. Berrani, C. Garcia, Contribution of recurrent connectionist language models in improving lstm-based arabic text recognition in videos, Pattern Recognition 64 (2017) 245–254.
[41] R. Wu, S. Yang, D. Leng, Z. Luo, Y. Wang, Random projected convolutional feature for scene text recognition, in: Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on, IEEE, 2016, pp. 132–137.
[42] Y. Yu, W. Pedrycz, D. Miao, Neighborhood rough sets based multi-label classification for automatic image annotation,

International Journal of Approximate Reasoning 54 (9) (2013) 1373–1387.
[43] M. J. Fonseca, J. A. Jorge, Using fuzzy logic to recognize geometric shapes interactively, in: Fuzzy Systems, 2000. FUZZ IEEE 2000. The Ninth IEEE International Conference on, Vol. 1, IEEE, 2000, pp. 291–296.
[44] S. Ghanei, K. Faez, Localizing scene texts by fuzzy inference systems and low rank matrix recovery model, Computer Vision and Image Understanding 142 (2016) 94–110.
[45] M.-K. Zhou, X.-Y. Zhang, F. Yin, C.-L. Liu, Discriminative quadratic feature learning for handwritten chinese character recognition, Pattern Recognition 49 (2016) 7–18.
[46] L. P. Z. Pawlak, A. Skowron, Rough set theory.
[47] V. K. Lamba, Neuro fuzzy system.

***Author Biography**

Sangheeta Roy is a Ph.D candidate at University of Malaya (UM), Malaysia. Her area of interest includes image processing, pattern recognition and video text understanding.

Palaiahnakote Shivakumara is a Senior Lecturer in Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia. Previously, he was with the Department of Computer Science, School of Computing, National University of Singapore from 2008-2013 as a Research Fellow on Video text extraction and recognition project. He received B.Sc., M.Sc., M.Sc Technology by research and Ph.D degrees in computer science respectively in 1995, 1999, 2001 and 2005 from University of Mysore, Karnataka, India. He has been serving as Associate Editor for ACM Transactions Asian and Low-Resource Language Information Processing (TALLIP). He has published more than 190 papers in conference and journals. His research interests are in the area of image processing and Video text analysis.

Namita Jain is a Visiting scientist at Machine Intelligence Unit, Indian Statistical Institute, Kolkata. Her area of interest includes pattern recognition, Machine learning, feature selection.

Vijeta Khare is Ph.D candidate at University of Malaya, Malaysia. She received her B. Tech degree in Computer Science & Engineering from Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, India, in 2006 and M. Tech degree from ABV-Indian Institute of Information Technology and Management, Gwalior, India in 2008. Her research interest includes image processing, pattern recognition and video text processing.

Anjan Dutta received his PhD in Computer Science from the Universitat Autònoma de Barcelona in the year of 2014. In his PhD he worked on inexact subgraph matching applied for symbol spotting in graphical documents. He received the Extraordinary PhD Thesis Award for the year 2013-14 by the Universitat Autònoma de Barcelona for outstanding dissertation. Before his PhD, he obtained MS in Computer Vision and Artificial Intelligence also from the Universitat Autònoma de Barcelona, MCA in Computer Applications from the West Bengal University of Technology and BS in Mathematics (Honors) from the University of Calcutta respectively in the year of 2010, 2009 and 2006. He worked as a postdoctoral researcher at a few academic institutes. He has published several papers in reputed journals. His recent research interests have revolved around graph-based representation for visual objects and graph-based algorithms for solving various tasks in Computer Vision, Pattern Recognition and Machine Learning.

Umapada Pal received his Ph.D. from Indian Statistical Institute and his Ph.D. work was on the development of Printed Bangla OCR system. He did his Post-Doctoral research on the segmentation of touching English numerals at INRIA (Institut National de Recherché en Informatiqueet en Automatique), France. From January 1997, he is a Faculty member of the Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata and at present he is a Professor. Because of his significant impact in the Document Analysis research domain of Indian language, TC-10 and TC-11 committees of IAPR (International Association for Pattern Recognition) presented 'ICDAR Outstanding Young Researcher Award' to

Dr. Pal in 2003. He is the Editorial board member of International Journal of Computer, Mathematical Sciences and Applications; Electronic Letters on Computer Vision and Image Analysis; and ACM Transactions on Asian Language Information Processing. He is a life member of IUPRAI (Indian unit of IAPR) and senior life member of Computer Society of India.

Tong Lu received the PhD degree in computer science from Nanjing University in 2005. He received his M.Sc. and B.Sc. degree from the same university in 2002 and 1993, respectively. He is now a full Professor at the same university. His current interests are in the areas of multimedia, computer vision and pattern recognition algorithms/systems.